The Linear Algebra of Block Quasi-Newton Algorithms

Dianne P. O'Leary*

Computer Science Department and
Institute for Advanced Computer Studies
University of Maryland
College Park, Maryland 20742

A. Yeremin

Institute of Numerical Analysis of the Russian Academy of Sciences
Leninskij Prospekt 32A
117 334 Moscow, Russia

Submitted by Leiba Rodman

ABSTRACT

The quasi-Newton family of algorithms for minimizing functions and solving systems of nonlinear equations has achieved a great deal of computational success and forms the core of many software libraries for solving these problems. In this work we extend the theory of the quasi-Newton algorithms to the *block* case, in which we minimize a collection of functions having a common Hessian matrix, or we solve a collection of nonlinear equations having a common Jacobian matrix. This paper focuses on the linear algebra: update formulas, positive definiteness, least-change secant properties, relation to block conjugate gradient algorithms, finite termination for quadratic function minimization or solving linear systems, and the use of the quasi-Newton matrices as preconditioners.

1. INTRODUCTION

Algorithms in the family of *block conjugate gradient methods* have been investigated for two distinct kinds of problems: first, for solving several linear systems involving the same matrix, and second, for solving a single linear system making use of more than one initial guess in order to change

*This work was supported by NSF Grant CCR 9115568. Email: `oleary@cs.umd.edu`.

655 Avenue of the Americas, New York, NY 10010 0024-3795/94/$7.00

the rate of convergence. In either case, a major motivation for the use of the algorithm is the interplay between the form of the matrix and the computer architecture: on many computer systems, significant savings arise in multiplying p vectors by large sparse matrices if the matrix is accessed and applied to all of the vectors at once rather than accessing the matrix multiple times and computing the products one at a time. The block algorithms can exploit this fact.

The *block quasi-Newton methods* can be approached from two similar viewpoints. First, consider the problem of finding the minimizers for the p functions

$$f_i(x) = \widehat{f}(x) + b_i^T x + c_i,$$

with $x \in \mathcal{R}^n$. For distinct values of the vectors b_i, each of these functions has a different minimizer x_i^* and gradient $g_i(x)$, but they share a common Hessian matrix $G(x)$. We would like to solve these problems, taking advantage of information that the solution process provides about the common Hessian matrix. Problems of this form arise, for example, in studying elasticity under various loading conditions.

For the second viewpoint, consider the problem of finding a minimizer of a single function $\widehat{f}(x)$. Given different initial guesses $x_1, x_2, \ldots, x_p$, we form the p functions

$$f_i(x) = \widehat{f}(x - x_i).$$

Again, the functions have different minimizers and gradient vectors but a common Hessian at the solution.

In this work we develop the linear algebra behind a block variable-metric (quasi-Newton) family of algorithms for solving these problem. Our development parallels in large part that in Dennis and Moré [7] for the standard quasi-Newton family of algorithms.

The update formulas we develop are closely related to the *multiple secant updates* developed in an unpublished technical report of Schnabel [11]. Byrd, Schnabel, and Schultz [4] discuss the parallel implementation of the resulting secant algorithm, and Byrd, Nocedal, and Schnabel [3] discuss limited-memory versions. Higham and Higham [8] show an alternative derivation of the least-change updates. Some of our results duplicate the original results of Schnabel; we include them here for completeness, and because several of the proof techniques are new. In the final section we outline the similarities and differences between the approaches.

Section 2 discusses the use of the block quasi-Newton algorithms for function minimization, including the double-rank update formulas, hereditary positive definiteness, inverse update formulas and updates to factorizations, finite termination and the relation to the block conjugate gradient family, and the convergence of Hessian approximations. Section 3 concerns the use of block quasi-Newton algorithms for solving nonlinear equations. The paper concludes with numerical examples and some final remarks.

2. THE BLOCK QUASI-NEWTON FAMILY FOR FUNCTION MINIMIZATION

We start with the problems

$$\min_x f_i(x), \qquad i = 1, \ldots, p,$$

where the functions $f_i : \mathcal{R}^n \to \mathcal{R}$ share a common Hessian matrix.

We will solve the problems using a block version of the quasi-Newton algorithm, storing and updating an approximation $B^{(k)}$ to the Hessian. The algorithm is initialized by taking an initial guess $x^{(0)} \in \mathcal{R}^{n\times p}$ (column i is the guess for the function f_i) and an initial symmetric positive definite approximation to the common Hessian, usually $B^{(0)} = I$. At each step we compute the p gradients $\nabla f(x^{(k)}) \in \mathcal{R}^{n\times p}$ and the p search directions $s^{(k)} \in \mathcal{R}^{n\times p}$. The kth step of the algorithm is:

1. Compute a matrix $\widehat{s}^{(k)}$ of search directions by solving
$$B^{(k)}\widehat{s}^{(k)} = -\nabla f(x^{(k)}) = -\big[\nabla f_1(x^{(k)}), \ldots, \nabla f_p(x^{(k)})\big].$$
2. Update each column of the approximate solution $x^{(k+1)} = x^{(k)} + s^{(k)}$, where $s^{(k)} = \widehat{s}^{(k)}\alpha^{(k)}$, and the parameters $\alpha^{(k)} \in \mathcal{R}^{p\times p}$ are determined by (approximate) minimization of each of the p functions over the p-dimensional subspace spanned by the columns of $s^{(k)}$.
3. Update the approximate Hessian: $B^{(k+1)}$ is a function of the current matrix $B^{(k)}$, the step $s^{(k)}$, and the change in gradients
$$y^{(k)} = \nabla f\big(x^{(k+1)}\big) - \nabla f\big(x^{(k)}\big).$$

To simplify notation, we will omit superscripts whenever possible, denoting the $n \times p$ matrices $x^{(k)}, y^{(k)}$, and $s^{(k)}$ and the square matrix $B^{(k)}$ by x, y, s, and B respectively, and denoting $x^{(k+1)}$ and $B^{(k+1)}$ by $\overline{x}$ and $\overline{B}$ respectively.

We require two key properties of a quasi-Newton algorithm for minimization:

1. The *secant condition*, the defining condition for this family of algorithms, $\overline{B}s = y$.

2. If B is symmetric, then $\overline{B}$ must be symmetric.

For $p > 1$, these two conditions are incompatible for general functions f [11], although they can both be achieved if the functions are quadratic with a common Hessian matrix.

2.1. *The Symmetric Double-Rank Updates*

We will derive a double-rank family of updates following a technique due to Powell. For quadratic minimization problems, $y = Gs$, where G is the symmetric Hessian matrix common to all of the functions, and thus we have the important property that $s^T y = y^T s$. Our derivation depends upon this property.

ASSUMPTION 1. $s^T y = y^T s$.

Note that we can compute a matrix satisfying the secant condition by setting

$$C_1 = B + (y - Bs)(c^T s)^{-1} c^T,$$

where $c \in \mathcal{R}^{n \times p}$ is any matrix such that $c^T s$ is invertible. For convenience, without loss of generality, we will assume that $c^T s = I$. Unfortunately, the matrix C_1 fails to be symmetric. We can restore symmetry by letting

$$C_2 = \frac{C_1 + C_1^T}{2},$$

and iterate this process: $C_0 = B$,

$$C_{2j+1} = C_{2j} + (y - C_{2j}s)c^T, \qquad C_{2j+2} = \frac{C_{2j+1} + C_{2j+1}^T}{2}.$$

Letting $w_{2j} = y - C_{2j}s$, we can derive a recurrence for the even terms:

$$C_{2j+2} = C_{2j} + \frac{w_{2j}c^T + c w_{2j}^T}{2}$$

and

$$w_{2j+2} = y - C_{2j+2}s = \tfrac{1}{2}(I - cs^T)w_{2j} \equiv \tfrac{1}{2}Pw_{2j}.$$

The last two expressions rely on Assumption 1. The eigenvalues of $\frac{1}{2}P$ are $\frac{1}{2}$ and 0, so

$$\sum_{j=0}^{\infty} w_{2j} = \sum_{j=0}^{\infty} \left(\tfrac{1}{2}P\right)^j w_0 = \left(I - \tfrac{1}{2}P\right)^{-1}(y - Bs).$$

This is enough to establish convergence of the sequence $\{C_{2j}\}$, since

$$C_{2j} = B + \sum_{k=0}^{j-1} \frac{w_{2k}c^T + cw_{2k}^T}{2},$$

so we have that

$$\lim_{j\to\infty} C_{2j} = B + \frac{\left(I - \frac{1}{2}P\right)^{-1} w_0 c^T + c\left[\left(I - \frac{1}{2}P\right)^{-1} w_0\right]^T}{2}.$$

It is easily verified that

$$\left(I - \tfrac{1}{2}P\right)^{-1} = 2\left(I - \tfrac{1}{2}cs^T\right),$$

so

$$\begin{aligned} \lim_{j\to\infty} C_{2j} &= B + \frac{2\left(I - \frac{1}{2}cs^T\right)w_0c^T + c\left[2\left(I - \frac{1}{2}cs^T\right)w_0\right]^T}{2} \\ &= B + w_0c^T + cw_0^T - \frac{cs^Tw_0c^T + cw_0^Tsc^T}{2} \equiv \overline{B}. \end{aligned} \tag{1}$$

Note that Assumption 1 is a necessary condition in order that the formula for $\overline{B}$, equation (1), satisfy the secant condition $\overline{B}s = y$, since

$$\overline{B}s = y + c\left[\frac{y^Ts - s^Ty}{2}\right]. \tag{2}$$

We have thus established the result analogous to Lemma 7.2 of Dennis and Moré [7]:

THEOREM 1. *Let B be symmetric, and assume that $c^Ts = I$. Then the sequence C_{2j} defined above with $C_0 = B$ converges to $\overline{B}$ given by* (1) *and $\overline{B}s = y$.*

If $p = 1$, this update formula satisfies an important minimization property (see Theorem 7.3 of Dennis and Moré [7]), and we now show that the formula for $p > 1$ has a similar property.

THEOREM 2. *Let B be symmetric, and assume that $c^Ts = I$. Let $M \in \mathcal{R}^{n\times n}$ be any nonsingular symmetric matrix such that $Mc = M^{-1}s$. Then $\overline{B}$ defined by* (1), *is the unique solution to the problem*

$$\min_{\hat{B}}\{\|\hat{B} - B\|_{M,F} : \hat{B} \text{ symmetric}, \hat{B}s = y\},$$

where $\|C\|_{M,F}$ denotes the Frobenius norm of the matrix MCM.

Proof. Suppose that $Mc = M^{-1}s = z$ and M is symmetric. Suppose that $\widehat{B}$ is any symmetric matrix such that $\widehat{B}s = y$. Let

$$\overline{E} = M(\overline{B} - B)M$$

and

$$\widehat{E} = M(\widehat{B} - B)M.$$

Now, $z^T z = I$, and using (1), we see that

$$\overline{E} = Mw_0 z^T + zw_0^T M - \frac{zs^T w_0 z^T + zw_0^T sz^T}{2},$$

and since

$$Mw_0 = M(y - Bs) = M(\widehat{B} - B)s = \widehat{E}z,$$

we have $w_0^T s = z^T \widehat{E} z$ and

$$\overline{E} = \widehat{E}zz^T + zz^T\widehat{E} - \frac{zz^T\widehat{E}zz^T + zz^T\widehat{E}zz^T}{2}.$$

It is easy to verify that

$$\overline{E}z = \widehat{E}z,$$

and, if $v \in \mathcal{R}^{n\times p}$ satisfies $v^T z = 0$, then $\overline{E}v = zz^T\widehat{E}v$, so

$$\|\overline{E}v\|_F = \|zz^T\widehat{E}v\|_F \le \|zz^T\|_2 \|\widehat{E}v\|_F \le \|\widehat{E}v\|_F.$$

Thus, $\overline{B}$ is the minimizer. ∎

2.2. *Hereditary Positive Definiteness*

We now establish conditions under which we can guarantee that if B is positive definite, then $\overline{B}$ is also positive definite.

The standard argument for the quasi-Newton algorithm with $p = 1$ is based on verifying that the determinant of $\overline{B}$ is positive, and noting that the update could have changed the sign of at most one eigenvalue. This argument fails for $p > 1$, so we adopt a different approach here.

From the definition (1) for $\overline{B}$, using the assumption that $s^T y = y^T s$, we have

$$\overline{B} \; = \; B + (y - Bs)c^T + c(y - Bs)^T - cs^T(y - Bs)c^T$$

$$
\begin{aligned}
&= B - Bsc^T - cs^T B + cs^T Bsc^T + yc^T + cy^T - cs^T yc^T \\
&= (I - cs^T)B(I - sc^T) + yc^T + cy^T - cs^T yc^T \\
&\equiv PBP^T + yc^T + cy^T - cs^T yc^T.
\end{aligned}
$$

For the block generalization of the Davidon-Fletcher-Powell (DFP) update formula, we have $c = y\tau$ for $\tau = (s^T y)^{-1}$, so

$$\overline{B} = PBP^T + y(s^T y)^{-1} y^T \tag{3}$$

is the sum of two symmetric positive semidefinite matrices. Consider $z^T \overline{B} z$. Both summands are nonnegative if z is nonzero, and, in fact, at least one of the terms is positive: if $P^T z = 0$, then the second term is positive, while if $P^T z$ is nonzero, the first is positive. Therefore we have proven the following result.

THEOREM 3. *The block-DFP update formula can be expressed as* (3) *with* $P = I - y(s^T y)^{-1} s^T$, *and if* B *and* $s^T y$ *are positive definite, then* B *is also positive definite.*

2.3. Inverse Update Formulas and Updates to Factors

Since it is necessary to solve linear systems involving the approximate Hessian matrix B, we need to have a representation that allows this to be performed economically. Typically, rather than storing the matrix B, we store $H \equiv B^{-1}$ or triangular factors of B.

A development similar to that of Section 2.1 can be carried out using the matrices H rather than B. The secant condition becomes

$$\overline{H} y = s.$$

We can conclude that the double-rank updates are members of the family

$$\overline{H} = H + zd^T + dz^T - \frac{dy^T zd^T + dz^T yd^T}{2}, \tag{4}$$

where $z = s - Hy$, and $d \in \mathcal{R}^{n \times p}$ is chosen so that $d^T y = I$.

The inverse updates also satisfy a minimization property:

THEOREM 4. *If* H *is symmetric,* $d^T y$ *is a positive definite matrix, and* $M \in \mathcal{R}^{n \times n}$ *is any nonsingular symmetric matrix such that* $Md = M^{-1} y$, *then* $\overline{H}$ *defined by* (4) *is the unique solution to the problem*

$$\min_{\widehat{H}} \{ \|\widehat{H} - H\|_{M,F} : \widehat{H} \text{ symmetric}, \widehat{H} y = s \}.$$

The block form of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update (i.e., the complementary block-DFP formula) results from the choice $d = s\tau$, where $\tau = (y^T s)^{-1}$. This update has the same hereditary positive definiteness properties as does block-DFP and can be expressed in the form

$$\overline{H} = \left[I - s\left(y^T s\right)^{-1} y^T\right] H \left[I - y(s^T y)^{-1} s^T\right] + s\left(s^T y\right)^{-1} s^T. \qquad (5)$$

The Cholesky factors LL^T for B can also be updated in a simple way, and update formulas can be derived that preserve the sparsity of L while achieving the secant condition. We leave the discussion of these updates to a future paper.

Various choices of c in the update of B and of d in the update of H result in block generalizations of some well-known algorithms:

B update	$c = s\tau$	block PSB method
	$c = y\tau$	block DFP method
H update	$d = y\tau$	block Greenstadt method
	$d = s\tau$	block BFGS method

2.4. Finite Termination and Relation to the Block Conjugate Gradient Algorithm

The block-BFGS method and the block DFP methods have the property of finite termination on quadratic functions, while the block Greenstadt and the PSB (like the original Greenstadt and PSB algorithms) do not. We will establish the property for the BFGS version in a proof analogous to that of Broyden [2].

Suppose we are minimizing p quadratic functions with common Hessian matrix G and linear coefficient vectors $b_1, \ldots, b_p$. Let C be a positive definite matrix satisfying $C^2 = G$. Define the $n \times p$ matrix e by

$$e = C(x - x^*), \qquad K = CHC,$$

where the ith column of x^* is the optimal solution to the ith minimization problem. In this notation, we have

$$\begin{aligned}
\nabla f(x) &= Gx - b = Ce, \\
s &= -HCe\alpha, \\
\alpha &= (e^T K^2 e)^{-1} e^T K e, \\
y &= Gs = -CKe\alpha, \\
\widehat{z} &\equiv Cz = -(Ke - K^2 e)\alpha,
\end{aligned}$$

$$
\begin{aligned}
\widehat{d} &\equiv Cd, \\
z^T y &= -\widehat{z}^T K e\alpha, \\
\widehat{d}^T K e &= -d^T y \alpha^{-1} = -\alpha^{-1}, \\
\overline{K} &= K + \widehat{z}\widehat{d}^T + \widehat{d}\widehat{z}^T + \widehat{d}\widehat{z}^T K e\alpha \widehat{d}^T, \\
\overline{e} &= e - K e\alpha.
\end{aligned}
$$

We see that

$$\overline{e}^T K e = (e^T - \alpha^T e^T K) K e = 0 \tag{6}$$

and

$$\overline{K} K e = K e \tag{7}$$

This sets up an induction argument. Assume that

$$
\begin{aligned}
e^{(i)^T} K^{(j)} e^{(j)} &= 0, && 1 \le j \le i-1, \\
K^{(i)} K^{(j)} e^{(j)} &= K^{(j)} e^{(j)}, && 1 \le j \le i-1.
\end{aligned}
$$

It is easy to show that

$$
\begin{aligned}
e^{(i+1)^T} K^{(j)} e^{(j)} &= 0, && 1 \le j \le i, \\
K^{(i+1)} K^{(j)} e^{(j)} &= K^{(j)} e^{(j)}, && 1 \le j \le i.
\end{aligned}
$$

It is in the proof of this last statement that we need to use the special form of the BFGS parameters: choosing $d = s\tau$ ensures that for $j < i, \widehat{d}^{(i)^T} K^{(j)} e^{(j)} = 0$. These equations, plus (6) and (7), complete the induction.

Now, noting that for $j < i$

$$
\begin{aligned}
e^{(i)^T} K^{(i)} K^{(j)} e^{(j)} = 0 &= e^{(i)^T} C H^{(i)} C C H^{(j)} C e^{(j)} \\
&= \alpha^{(i)^{-1}} s^{(i)^T} G s^{(j)} \alpha^{(j)^{-1}},
\end{aligned}
$$

we have established conjugacy of the search directions. Thus if the search directions are linearly independent (i.e., if the p columns of each matrix $s^{(k)}$ are linearly independent), then the algorithm must terminate in at most $\lceil n/p \rceil$ iterations. The algorithm minimizes the same quantities as the block conjugate gradient algorithm, and if $H^{(0)} = I$, then it minimizes over the same subspace as the block conjugate gradient algorithm, and the iterates must be identical.

The proof for finite termination of the block-DFP method and conjugacy of the directions is similar, using the representation

$$\overline{H} = H - Hy(y^T H y)^{-1} y^T H + s(s^T y)^{-1} s^T$$

for the inverse of the block-DFP matrix $\overline{B}$.

THEOREM 5. *The block-BFGS and the block-DFP algorithms terminate with the optimal solution in at most $\lceil n/p \rceil$ iterations when applied to a quadratic function, assuming that the search directions remain linearly independent.*

2.5. *Convergence of the Hessian Approximations*

It is known that, for quadratic functions, the sequence $\{B_k\}$ for any members of the so-called Broyden family of updates (including the BFGS and the DFP algorithms) converges to the Hessian matrix G on a subspace corresponding to the search directions. Thus, if the algorithm does not terminate prematurely, after n iterations B is equal to G. The block-Broyden family seems to have similar properties. Here we establish the result only for the block-BFGS update.

The property arises from using (5) and the fact that $y = Gs$ to derive the formula

$$\overline{H} - G^{-1} = \widehat{P}(H - G^{-1})\widehat{P}^T,$$

where $\widehat{P} = I - s(s^T G s)^{-1} y^T$. We see that

$$(\overline{H} - G^{-1})Gs = 0,$$

and since the search directions s are conjugate, $\widehat{P}^{(i)^T} s^{(j)} = 0$ for $j \le i$.

A similar result holds for the sequence $\{B^{(k)}\}$ as an approximation to G when the block-DFP update is used. The argument parallels the one above.

THEOREM 6. *For the block-BFGS algorithm, the matrix $H^{(k)}$ agrees with G^{-1} on the subspace spanned by $\{Gs^{(0)}, \ldots, Gs^{(k-1)}\}$ for $k = 1, 2, \ldots$. For the block-DFP algorithm, the matrix $B^{(k)}$ agrees with G on the subspace spanned by $\{s^{(0)}, \ldots, s^{(k-1)}\}$ for $k = 1, 2, \ldots$.*

3. SOLVING NONLINEAR EQUATIONS USING BLOCK QUASI-NEWTON ALGORITHMS

Block quasi-Newton algorithms can also be derived for the solution to linear or nonlinear equations

$$g_i(x) = 0, \qquad i = 1, \ldots, p,$$

where the functions $g_i : \mathcal{R}^n \rightarrow \mathcal{R}^n$, share a common Jacobian matrix $J(x)$. The secant condition that $\overline{B}s = y$ remains, but the requirement of symmetry is removed.

The block-Broyden ("good") update

$$\overline{B} = B + (y - Bs)(s^T s)^{-1} s^T \tag{8}$$

satisfies a minimization property:

THEOREM 7. *Given $B \in \mathcal{R}^{n\times n}$, $y \in \mathcal{R}^n$, and a nonzero $s \in \mathcal{R}^n$, the $\overline{B}$ defined by* (8) *is the unique solution to the problem*

$$\min_{\hat{B}}\{\|\hat{B} - B\|_F : \hat{B}s = y\}.$$

Other update formulas are of the form

$$\overline{B} = B + (y - Bs)c^T,$$

where v satisfies $c^T s = I$.

As before, the algorithm can be implemented by updating the inverse of B:

$$\overline{H} = H + (s - Hy)d^T,$$

where d is chosen so that $d^T y = I$. There are two common choices for c and d:

Method	B update	H update
Block-Broyden "good" method	$c = s\tau$	$d = H^T s\tau$
Block-Broyden "bad" method	$c = B^T y\tau$	$d = y\tau$

The algorithm can also be implemented by updating factors of B. In this case, the QR factors are to be preferred to the LU.

In contrast to the symmetric updates, the Jacobian approximations $B^{(k)}$ do not converge to the Jacobian matrix. For simplicity, we will assume that $2n/p$ is an integer. The argument exactly parallels that in [10], and it can be shown that if G is constant, independent of x (i.e., each $g_i(x) = 0$ is a linear system of equations), then the matrix

$$F^{(k)} \equiv I - GH^{(k)}$$

has rank $n-p$ as long as:

1. $d^{(j)^T} y^{(j-1)}$ has no zero columns, $j = 1, 2, \ldots, k$, and $y^{(k)}$ has full rank.
2. $d^{(0)}$ is in the range of $F^{(0)^T}$.
3. $k \leq 2n/p$ is odd.

Thus, the approximations to G always differ on a subspace of dimension $n-p$.

The algorithm does possess finite termination when used to solve linear systems, however. Let

$$P^{(k)} = F^{(k)} F^{(k-1)} \cdots F_0 = F^{(k)} P^{(k-1)}.$$

Then

$$\begin{aligned} g^{(k+1)} &= P^{(k)} g^{(0)}, \\ y^{(k)} &= -G H^{(k)} P^{(k-1)} g^{(0)}, \\ s^{(k)} &= -H^{(k)} P^{(k-1)} g^{(0)}. \end{aligned}$$

Thus, the character of the product matrices $P^{(k)}$ determines the behavior of the residuals $g^{(k)}$ in the course of the iteration. The key to this behavior is the nature of the left null vectors of $P^{(k)}$, the vectors z for which $z^T P^{(k)} = 0$. In particular, the factor matrices $F^{(k)}$ are *defective*, having p Jordan blocks of size $\lceil k/2 \rceil$ corresponding to zero eigenvalues. The linearly independent left *principal vectors* of $F^{(k)}$, the columns of $z_i \in \mathcal{R}^{n \times p}$ satisfying

$$\begin{aligned} z_1^T F^{(k)} &= 0, \\ z_3^T F^{(k)} &= z_1^T, \\ z_i^T F^{(k)} &= z_{i-2}^T, \end{aligned}$$

do not depend on k, and in fact each column is a left *eigenvector* of $P^{(k)}$ corresponding to a zero eigenvalue.

THEOREM 8. *The vector $g^{(k+1)}$ is orthogonal to the space spanned by the columns of the matrices $\{z_i\}$ (i odd and $i \leq k$) and, after at most $2n/p$ steps, is forced to be zero.*

Proof. The argument parallels that in [10]. ∎

4. SOME NUMERICAL EXAMPLES

We concentrate here on solving linear systems of equations, leaving the application to function minimzation and nonlinear equations to future work. Our motivation is to study the use of the block quasi-Newton matrices to generate preconditioners for other nearby problems.

Our example problems are related to finding stationary vectors of Markov chains arising in overflow queuing networks. We consider the problem in null-vector form

$$Gz = 0$$

For each queue, we specify arrival and service rates, and overflow is allowed from each queue to its successor. As noticed by Kaufman [9] and Chan [5], there are many analogies between these systems and those arising from discretization of elliptic partial differential equations. The resulting matrix has a regular sparsity structure: a five-point operator for two queues, a seven-point operator for three queues, etc. If overflow is eliminated, then the problem is separable, and the matrix is a tensor product, for which linear systems can easily be solved.

The matrix is determined by the specification of a relatively small number of parameters: the arrival and service rates. Often parametric studies are desirable, in which, for example, the behavior is studied for an entire range of parameters. In order to develop efficient algorithms for these parametric studies, we propose using the block quasi-Newton algorithms with p different initial guesses to solve the problem $Gz = 0$ for one set of parameters, and use the resulting H matrix and z vectors as initial guesses for other problems.

We find the stationary vector using three algorithms, using the separable problem as a preconditioner for each [5]. This reduces the problem to an identity matrix plus a low-rank correction, and through a *capacitance matrix technique*, we iterate only on the low-rank problem of dimension n much less than the number of states. The algorithms are:

1. The block-BFGS algorithm on the problem symmetrized by multiplying by the transpose operator.
2. The block-Broyden algorithm on the nonsymmetric problem.
3. The single-vector Arnoldi algorithm.

The test problems were three-, four-, and five-queue problems with arrival rates of $\lambda_1 = 0.9$, $\lambda_2 = 0.7$, $\lambda_3 = \lambda_4 = \lambda_5 = 0.5$ and service rates of $\mu_1 = 0.1$, $\mu_2 = 0.4$, $\mu_3 = \mu_4 = \mu_5 = 0.5$. We solved both the original problem and a perturbed problem formed by modifying the arrival and service rates by random perturbations uniformly distributed on [0, 0.1].

TABLE 1. Number of Iterations for Various Algorithms

Problem			Number of iterations				
Queues	States	Unknowns	BFGS-2	Broyden-2	BFGS-1	Broyden-1	Arnoldi-1
3	252	102	16	13	21	15	13
No restart			6	8	7	10	13
Smart restart			14	9	16	9	11
4	144	108	15	11	20	13	11
No restart			6	7	7	8	11
Smart restart			11	6	12	7	9
5	108	100	17	11	20	12	11
No restart			5	6	9	8	11
Smart restart			12	5	13	4	8
4	500	308	24	15	28	17	14
No restart			8	11	10	14	14
Smart restart			16	9	19	9	11
5	360	312	24	15	30	16	13
No restart			6	10	8	11	13
Smart restart			13	7	15	5	9

For the block-BFGS and block-Broyden algorithms, there were two ways to solve the perturbed problem:

1. Save the H matrix and the stationary vector for the original problem and reuse them ("no restart").
2. Reuse only the stationary vector for the original problem ("smart restart").

The iterations were halted when the norm of the residual was less than 10^{-4}. The result of the experiment are shown in Table 1. Saving the Broyden matrix is often worse than discarding it and using the identity matrix, as we might predict from the results in Section 3, since the Broyden algorithm does not terminate with $H = G^{-1}$. On the other hand, using the BFGS matrix as a preconditioner can sometimes save as much as 2/3 of the iterations, because this matrix is an increasingly good approximation to G^{-1}.

5. CONCLUDING REMARKS

We have derived block forms of variable-metric algorithms for solving nonlinear equations and unconstrained optimization problems, and demonstrated their use for the special case of solving symmetric linear systems.

Further work needs to be done in making the algorithms useful for general nonlinear functions.

One interesting application of the method is in deriving preconditioners for linear systems of equations; a few steps of the block algorithm produce an approximation to the inverse of the Hessian matrix, and this approximation is the original guess plus a low-rank correction. Such preconditioners can be quite useful in solving additional linear systems involving the same matrix or a similar one.

The work of Schnabel [11] is closely related to the work presented here, but has quite different motivation. In place of our p new search directions $s^{(k)}$ used in the secant condition, he used the latest p changes $s^{(k)}$, $s^{(k-1)}, \ldots, s^{(k-p+1)}$, where each s is a single vector. He determined update formulas for Broyden's "good" method, the Powell symmetric Broyden method, DFP, and BFGS, and showed least-change secant properties. He also showed a q-superlinear convergence rate under the usual conditions and discovered that preserving positive definiteness, symmetry, and the secant condition are generally incompatible—possible if and only if $y^T s$ is symmetric positive definite. Schnabel note that the methods generalize the "projected" updates of Davidon [6] and a method of Barnes [1].

Further work was done in Byrd, Schnabel, and Shultz [4], which gives a short discussion of parallel implementation, and in Byrd, Nocedal, and Schnabel [3], which discusses limited-memory versions.

It should be noted that the method of Schnabel *et al.* does not give the n/p- or $2n/p$-step termination on linear problems achieved by the block quasi-Newton family.

The block quasi-Newton methods are quite appropriate for parallel computation. They replace the vector-vector products in the overhead of the quasi-Newton algorithm by a smaller number of matrix-vector products, permitting greater utilization of vector pipelines and fewer global synchronization points. They also induce parallelism in the function evaluations and minimizations over p-dimensional subspaces. For linear problems, several matrix-vector products by G are performed each iteration, increasing the amount of computation relative to memory traffic. For general problems, the function evaluations can be done independently, as can the minimization of each of the p functions over the subspaces.

We are grateful to Tamara Gibson and the referees for their helpful comments.

REFERENCES

1 J. Barnes, An algorithm for solving nonlinear equations based on the secant method, *Comput. J.* 8:66–72 (1965).
2 C. G. Broyden, The convergence of a class of double-rank minimization algorithms 1, General considerations, *J. Inst. Math. Appl.* 6:76–90 (1970).
3 Richard H. Byrd, Jorge Nocedal, and Robert B. Schnabel, Representations of Quasi-Newton Matrices and Their Use in Limited Memory Methods, Technical Report CU-CS-612-92, Dept. of Computer Science, Univ. of Colorado at Boulder, 1992.
4 Richard H. Byrd, Robert B. Schnabel, and Gerald A. Schultz, Parallel quasi-Newton methods for unconstrained optimization, *Math. Programming* 42:273–306 (1988).
5 R. H. Chan, Iterative methods for overflow queueing models I, *Numer. Math.* 51:143–180 (1987).
6 W. C. Davidon, Optimally conditioned optimization algorithms without line searches, *Math. Programming* 9:1–30 (1975).
7 J. E. Dennis and Jorge J. Moré, Quasi-Newton methods, motivation and theory, *SIAM Rev.* 19:46–89 (1977).
8 Desmond J. Higham and Nicholas J. Higham, Componentwise perturbation theory for linear systems with multiple right-hand sides. *Linear Algebra Appl.* 174:111–129 (1992).
9 Linda Kaufman, Matrix methods for queuing problems, *SIAM J. Sci. Statis. Comput.* 4:525–552 (1983).
10 Dianne P. O'Leary, Why Broyden's nonsymmetric method terminates on linear equations, *SIAM J. Optim.*, to appear.
11 Robert B. Schnabel, Quasi-Newton Methods Using Multiple Secant Equations, Technical Report CU-CS-247-83, Dept. of Computer Science, Univ. of Colorado at Boulder, 1983.

Received 23 June 1993; final manuscript accepted 23 November 1993